MILLENNIUM EDITION

ORIGINAL

New updated CD with interactive computer tour

HOW COMPUTERS WORK

NATIONAL BESTSELLER
MORE THAN 1 MILLION COPIES SOLD

by Ron White

Illustrated by Timothy Edward Downs

How Computers Work

Millennium Edition

01-19-01A11:43 RCVD

Ron White

Illustrated by Timothy Edward Downs and Stephen Adams

que®

A Division of Macmillan Computer Publishing, USA
201 W. 103rd Street
Indianapolis, IN 46290

How Computers Work, Millennium Edition

First printing: September 1999

Publisher	Greg Wiegand
Executive Editor	Angela Wethington
Acquisitions Editor	Angelina Ward
Development Editor	Sarah Robbins
Managing Editor	Thomas F. Hayes
Project Editor	Leah Kirkpatrick
Copy Editor	Leah Kirkpatrick
Technical Illustrators	Timothy Edward Downs and Stephen Adams
Book Designers	Anne Jones and Dan Armstrong
Page Layout	Trina Wurst
Proofreader	Tricia Sterling
Indexer	Christine Nelsen

Manufactured in the United States of America

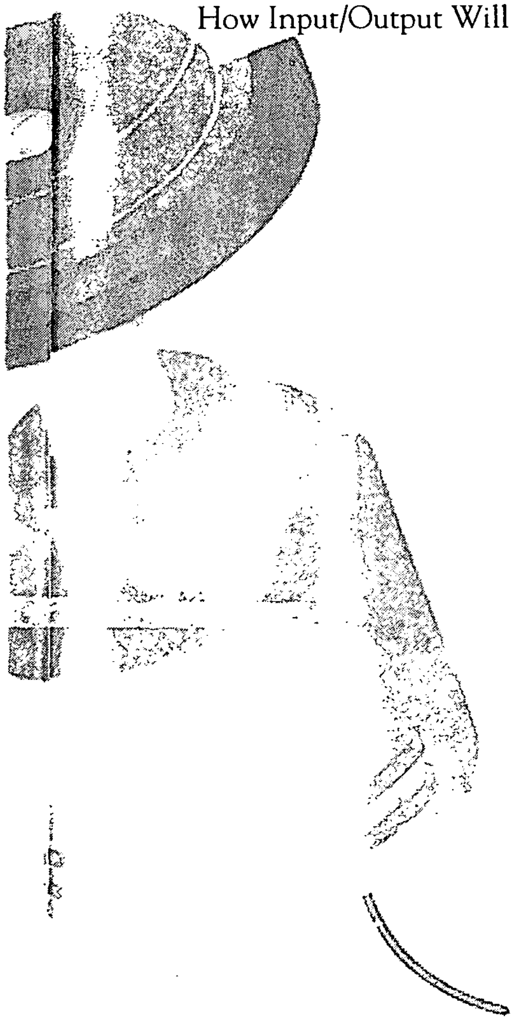
Library of Congress Catalog No.: 98-85290

ISBN: 0-7897-2112-0

00 6

Interpretation of the printing code: the rightmost double-digit number is the year of the book's printing; the rightmost single-digit number, the number of the book's printing. For example, a printing code of 99-1 shows that the first printing of the book occurred in 1999.

This book was written on a Hewlett-Packard 7370V, a Gateway Solo 2100, and an IBM Aptiva Pentium II using WordPerfect, Microsoft Word, Professional Capture System, Paint Shop Pro, Notes 4.5, Windows Explorer, Netscape Navigator, and a BitSURFR ISDN connection. Author's color proofs produced with Hewlett-Packard OfficeJet Pro 1150C. It was produced on a Power Macintosh G3, with the following applications: Adobe Illustrator, QuarkXPress, Microsoft Word, Adobe Photoshop, Aldus FreeHand, and Collage Plus.



PERSONAL computers have changed dramatically since the first edition of this book in 1993. Then, 5-1/4-inch floppy drives were still common, an Intel 80386 was a hot processor, and it was still a toss-up to see whether Windows or some other operating system would dominate PCs. Really. What wasn't even mentioned in the first edition and merited only a couple of chapters in the second edition—the Internet—has become an entire section that grows yearly. Many of the topics covered in this edition were not dreamed of back then—Internet video and audio, rewriteable DVD drives, accelerated 3D graphics, the universal serial bus, global satellite positioning, and computers—real ones, not toys—that fit in your shirt pocket and let you write on them. To the extent back then that something like speaking to your computer had been imagined, it was still something of science fiction, something for tomorrow. Well, tomorrow's here.

The change in PCs has been so rapid, so extensive, and so detailed, I could never have learned the tricks behind the new technologies without the help of a lot of knowledgeable people who were not only generous with that knowledge, but patient enough to explain it to me until the mental LED lit up.

I had the privilege of launching the *How It Works* feature in *PC Computing* in 1989, but over the years, many people have worked on the series, and several chapters are based on the research and explanations they've done. Deep thanks go to Margaret Ficklen, Herb Brody, Brett L. Glass, Marty Jerome, Raymond Jones, Matthew Lake, Liesl La Grange Noble, Jack Nimersheim, Randy Ross, Stephen Sagman, Jan Smith, Dylan Tweney, Doug van Kirk, Mark L. Van Name and Bill Catchings, Christine Grech Wendin, and Kenan Woods. Thanks to Preston Gralla for too many things to list.

I'm also grateful to the dozens of people in the PC industry who've shared their knowledge, schematics, and white papers to infuse *How Computers Work* with detail and accuracy. I'm particularly grateful to Joe Vanderwater, Bryan Langerdoff, Dan Francisco, [illegible] Tami Casey, John Hyde, Bill Kircos, Susan Shaw, and Seth Walker of Intel; Russell Sanchez and Adam Kahn of Microsoft; Jim Bartlett, Ellen Reid Smith, Tim Kearns, and Desire Russell for IBM, Barrett Anderson and Todd-Hollinshead at id Software; Marcie Pedrazzi of Adaptec; Eileen Algaze at Rockwell; A.J. Rodgers, Jennifer Jones, and Susan Bierma of Tektronix; Ben Yoder and Lisa Santa Anna of Epson; Dewey Hou of TechSmith Corporation; Ray Soneira, author of Sonera Technologies' DisplayMate, Kathryn Brandberry and Doyle Nicholas of Thrustmaster; Dani Johnston of Microtek Lab; Lisa Tillman of The Benjamin Group; Dvorah Samansky for Iomega, Brandon Talaich of SanDisk, Chris Walker with Pioneer USA, Carrie Royce of Cirque Corporation, Andy Marken of Marken Communications, and Tracy Laidlaw of Beyond Words. If there are any mistakes in this book, I can't lay the blame on them.

In many ways, writing this edition of *How Computers Work* has been made much easier by the Internet. I didn't keep a complete list of every place on the Net I found useful information, but a couple of sites were particularly outstanding: Dictionary of PC Hardware and Data Communications Terms by Mitchell Shnier at `http://www.ora.com/reference/dictionary/tsearch.cgi` and the Free On-line Dictionary of Computing maintained by Paul Mayer at `http://wagner.princeton.edu/foldoc/index.html`. These and more are found in links on the book's Web page `www.howcomputerswork.net`.

And thanks to the many staff members at *PC Computing* who shared their knowledge and time and to former PCC Publisher, Mike Edelhart, who started the ball rolling on this book. A special thanks to Jim Seymour and Nora Georgas for starting me on the road to excitement and danger that dog the professional computer journalist. At Que to John Pierce, Angie Wethington, Juliet Langley, Cindy Hudson, Melinda Levine, Lysa Lewallen, Sarah Robbins, and Leah Kirkpatrick for pulling everything together. Thanks to John Rizzo for tech editing and researching several chapters. And thanks to my wife, Sue, for her constant encouragement, spirit-raising, and humor while waiting for me to pop out of writing mode.

I learned long ago that a writer's skill depends largely on how well the writer can steal from others. In addition to just about every book in the Que *How It Works* series, four other books were invaluable for details on PC innards: *Understanding I/O Subsystems* by W. David Schwaderer and Andrew W. Wilson, Jr.; *Inside the IBM PC* by Peter Norton; *The PC Configuration Handbook* by John Woram; and *The Winn Rosch Hardware Bible* by Winn L. Rosch. Also helpful was *The Way Things Work* by David Macaulay, not only for its informative explanations of computers, but for its inspiring examples of how to combine text and art into clear explanations.

Finally, this book would not be what it is without the artwork of Timothy Edward Downs and Stephen Adams. Invariably, they transformed my Neanderthal sketches into clear, informative illustrations, but also managed to make them into wonderful works of art. As the original artist for the book and in his continuing work at *PC Computing*, Tim especially has created a new, visual way to communicate technology.

Any sufficiently advanced technology is indistinguishable from magic.

—*Arthur C. Clarke*

SORCERERS have their magic wands—powerful, potentially dangerous tools with a life of their own. Witches have their familiars—creatures disguised as household beasts that could, if they choose, wreak the witches' havoc. Mystics have their golems—beings built of wood and tin brought to life to do their masters' bidding.

We have our personal computers.

PCs, too, are powerful creations that often seem to have a life of their own. Usually they respond to a seemingly magical incantation typed at a C:> prompt or to a wave of a mouse by performing tasks we couldn't imagine doing ourselves without some sort of preternatural help. But even as computers successfully carry out our commands, it's often difficult to quell the feeling that there's some wizardry at work here.

And then there are the times when our PCs, like malevolent spirits, rebel and open the gates of chaos onto our neatly ordered columns of numbers, our carefully wrought sentences, and our beautifully crafted graphics. When that happens, we're often convinced that we are, indeed, playing with power not entirely under our control. We become sorcerers' apprentices, whose every attempt to right things leads to deeper trouble.

Whether our personal computers are faithful servants or imps, most of us soon realize there's much more going on inside those silent boxes than we really understand. PCs are secretive. Open their tightly sealed cases and you're confronted with poker-faced components. Few give any clues as to what they're about. Most of them consist of sphinx-like microchips that offer no more information about themselves than some obscure code printed on their impenetrable surfaces. The maze of circuit tracings etched on the boards is fascinating, but meaningless, hieroglyphics. Some crucial parts, such as the hard drive and power supply, are sealed with printed omens about the dangers of peeking inside, omens that put to shame the warnings on a pharaoh's tomb.

This book is based on two ideas. One is that the magic we understand is safer and more powerful than the magic we don't. This is not a hands-on how-to book. Don't look for any instructions for taking a screwdriver to this part or the other. But perhaps your knowing more about what's going on inside all those stoic components makes them all a little less formidable when something does go awry. The second idea behind this book is that knowledge, in itself, is a worthwhile and enjoyable goal. This book is written to respond to your random musings about the goings-on inside that box that you sit in front of several hours a day. If this book puts your questions to rest—or raises new ones—it will have done its job.

At the same time, however, I'm trusting that knowing the secrets behind the magician's legerdemain won't spoil the show. This is a real danger. Mystery is often as compelling as knowledge. I'd hate to think that anything you read in this book takes away that sense of wonder you have when you manage to make your PC do some grand, new trick. I hope that, instead, this book makes you a more confident sorcerer.

Before You Begin

This book has been written with a certain type of personal computer in mind—the "Wintel," a PC most often built around an Intel processor and running Microsoft Windows. Many of the specifics in these explanations apply only to that class of computer and those components. For Mac users, I suggest John Rizzo's *How the Mac Works*, and that you do some serious thinking about switching.

In more general terms, the explanations also may apply to Macintosh computers, UNIX workstations, and even minicomputers and mainframes. But I've made no attempt to devise universal explanations of how computers work. To do so would, of necessity, detract from the understanding that comes from inspecting specific components.

Even so, there is so much variety even within the Intel/Microsoft world of PCs that, at times, I've had to limit my explanations to particular instances or stretch the boundaries of a particular situation to make an explanation as generic as possible. If you spot anything that doesn't seem quite right in this book, I hope that my liberties with the particulars is the only cause.

Ron White
San Antonio, Texas

"I think there is a world market for maybe five computers."
—Thomas Watson, chairman of IBM, 1943

BEFORE your personal computer is turned on, it is a dead collection of sheet metal, plastic, metallic tracings, and tiny flakes of silicon. When you push the On switch, one little burst of electricity—only about 3–5 volts—starts a string of events that magically brings to life what otherwise would remain an oversized paperweight.

Even with that spark of life in it, however, the PC is still stupid at first. It has some primitive sense of self as it checks to see what parts are installed and working, like those patients who've awakened from a coma and check to make sure that they have all their arms and legs and that all their joints still work. But beyond taking inventory of itself, the newly awakened PC still can't do anything really useful, certainly nothing we would even remotely think of as intelligent.

At best, the newly awakened PC can search for intelligence—intelligence in the form of an operating system that gives structure to the PC's primitive, amoebic existence. Then comes a true education in the form of application software—programs that tell the PC how to do tasks faster and more accurately than we could. The PC becomes a student who has surpassed its teacher.

But not all kinds of computers have to endure such a torturous rebirth each time they're turned on. You encounter daily many computers that spring to life fully formed at the instant they're switched on. You may not think of them as computers, but they are: calculators, your car's electronic ignition, the timer in the microwave, and the unfathomable programmer in your VCR. The difference between these and the big box on your desk is hard wiring. Computers built to accomplish only one task—and they are efficient about doing that task—are hard-wired. But that means they are more like idiot savants than sages.

What makes your PC such a miraculous device is that each time you turn it on, it is a *tabula rasa* (blank slate), capable of doing anything your creativity—or, more usually, the creativity of professional programmers—can imagine for it to do. It is a calculating machine, an artist's canvas, a magical typewriter, an unerring accountant, and a host of other tools. To transform it from one persona to another merely requires setting some of the microscopic switches buried in the hearts of the microchips, a task accomplished by typing a command or by clicking with your mouse on some tiny icon on the screen.

Such intelligence is fragile and short-lived. All those millions of microscopic switches are constantly flipping on and off in time to dashing surges of electricity. All it takes is an errant instruction or a stray misreading of a single switch to send this wonderful golem into a state of catatonia. Or press the Off switch and what was a pulsing artificial life dies without a whimper.

Then the next time you turn it on, birth begins all over again.

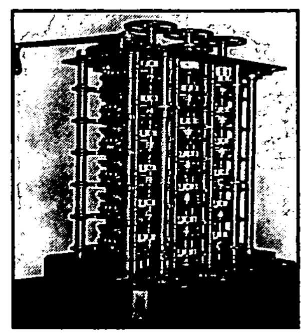
How Computers Used to Work

At the end of the 20th century, computers are such complex beasts—despite their relative youth—that it's difficult to imagine how such elaborate contraptions could have sprung fully grown from the brows of their creators. The explanation is, of course, that they didn't. The development of computers has been an evolutionary process and often it's well nigh impossible to figure out which came first, the chicken of software or the egg of hardware.

Human attempts to create tools to manipulate data date back at least as far as 2600 B.C. when the Chinese came up with the abacus. Leonardo da Vinci created a mechanical calculator. When the slide rule was invented in 1621, it remained the mathematician's tool of choice until the electronic calculator took over in the early 1970s.

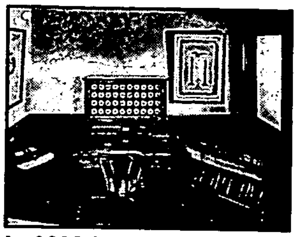
All the early efforts to juggle numbers had two things in common: They were mechanical and they were on a human scale. They were machines made of parts big enough to assemble by hand. Blaise Pascal's Arithmetic Machine used a system of gears turned by hand to do subtraction and addition. It also used punch cards for storing data, a method that's survived well into the 20th century.

In 1830 Charles Babbage invented—on paper—the Analytical Engine, which was different from its predecessors because, based on the results of its own computations, it could make decisions; such as sequential control, branching, and looping. But Babbage's machine was so complex that he died in 1871 without finishing it. It was built between 1989 and 1991 by dedicated members of the Science Museum in London. It was the physical size and complex mechanics of these mechanisms that limited their usefulness; they were only good for a few tasks, and they were not something that could be mass produced.



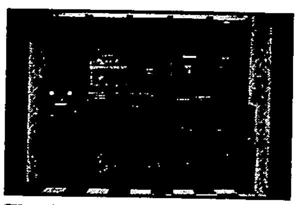
This portion of the Difference Engine #1, a forerunner to Charles Babbage's Analytical Engine—the first true computer—was completed in 1821. It contained 2,000 handmade brass parts. The entire machine would have used 25,000 parts and would have weighed 3 tons. The Analytical Engine was never completed, although part of it was built by Babbage's son, Henry, in 1910, and was found to be "buggy."

Courtesy of IBM



In 1888 Herman Hollerith, the founder of what was to become IBM, created a machine that used punched cards to tabulate the 1890 U.S. Census. The device tabulated the results in six weeks instead of the seven years it had taken to compile the census by hand.

Courtesy of Smithsonian Institution



The Eniac, built between 1943 and 1945, was the first all-electronic computer. It used so much power that legend says the lights of surrounding Philadelphia dimmed when the Eniac was switched on.

Courtesy of Smithsonian Institution

Mechanical devices of all types flourished modestly during the first half of the 20th century. Herman Hollerith invented a mechanized system of paper cards with holes in them to tabulate the U.S. census. Later, in 1924, Hollerith's Computing-Tabulating-Recording Company changed its name to International Business Machines.

Although no one could have known it at the time, the first break-through to the modern computer was in 1904 when John Ambrose Fleming created the first commercial diode vacuum tube, something Thomas Edison had discovered and discarded as worthless. The significance of the vacuum tube is that it was the first step beyond the human scale of machines. Until it came along, computations were made first by gears and then by switches. The vacuum tube could act as a switch turning on and off thousands of times faster than mechanical contraptions.

Vacuum tubes were at the heart of Colossus, a computer created by the British during World War II to break the codes produced by the German's Enigma encrypting machine. And the Germans reportedly came up with a general purpose computer—one not limited to a specific task as Colossus was. But the German invention was lost or destroyed in the war.

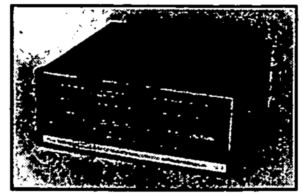
The war also gave birth to ENIAC (Electronic Numerical Integrator Analyzer and Computer), built between 1943 and 1945 by the U.S. Army to produce missile trajectory tables. ENIAC performed 5,000 additions a second although a problem that took two seconds to solve required two days to set up. ENIAC cost $500,000, weighed 30 tons, and was 100 feet long and 8 feet high. It contained 1,500 relays and 17,468 vacuum tubes.

Those same tubes that made ENIAC possible in the first place, were also its Achilles' heel. Consuming 200 kilowatts of electricity each hour, the tubes turned the computer into an oven, constantly cooking its own components. Breakdowns were frequent. What was needed was something that did the job of the tubes without the heat, bulk, and fragility. And that something had been around since 1926.

In 1926 the first semiconductor transistor was invented, but it wasn't until 1947, when Bell Labs' William Shockley patented the modern solid-state, reliable transistor, that a new era in computing dawned. The transistor did essentially the same thing a vacuum tube did—control the flow of electricity—but it was the size of a pea and generated little heat. Even with the transistor, the few computers built then still used tubes. It wasn't until 1954, when Texas Instruments created a way to produce silicon

transistors commercially, that the modern computer took off. That same year IBM introduced the 650, the first mass-produced computer. Business and the government bought 120 of them the first year.

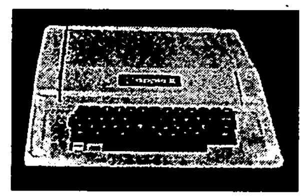
Four years later, Texas Instruments built the first integrated circuit by combining five separate components and the circuitry connecting them on a piece of germanium half an inch long. The integrated circuit led to the modern processor and has made a neverending contribution to smaller and smaller computers.



The first computer cheap enough for individuals to afford was the Altair 8800, created by a small New Mexico firm, MITS. It cost $367 without a keyboard or screen.

Courtesy of The Computer Museum

The computer grew increasingly smaller and more powerful, but its cost, complexity, and unswerving unfriendliness kept it the tool of a technological elite. It wasn't until 1975 that something resembling a personal computer appeared. The January issue of *Popular Electronics* featured on its cover something called the Altair 8800, made by Micro Instrumentation and Telemetry Systems (MITS). For $367 customers got a kit that included an Intel 8080 microprocessor and 256 bytes of memory. There was no keyboard; programs and data were both entered by clicking switches on the front of the Altair. There was no monitor. Results were read by interpreting a pattern of small red lights. But it was a real computer cheap enough for anyone to afford. MITS got orders for 4,000 within a few weeks.



The Apple, introduced in 1976, was an immediate hit partially because a program called Visicalc, which did the math of an electronic ledgersheet, justified the computer as a business cost.

Courtesy of Apple Corp.

The new computer was at first a toy for hobbyists and hackers. They devised clever ways to expand the Altair and similar microcomputers with keyboards, video displays, magnetic tape, and then diskette storage. Two hackers—Stephen Jobs and Steve Wozniak—created a personal computer that came complete with display, built-in keyboard, and disk storage, and began hawking it at computer clubs in California. They called it the Apple, and it was the first personal computer that was powerful enough, and friendly enough, to be more than a toy. The Apple, along with computers from Radio Shack and Commodore, began appearing in businesses, sometimes brought in behind the backs of the people in white lab coats who ran the "real," mainframe computers in a sealed room down the hall. The information services—or IS, as the professional computer tenders came to be called—disparaged the new computers as toys and at the same time they saw microcomputers as a threat to their turf.

The development that finally broke the dam, unleashing microcomputers on a society that would forever after be different, was not a technical invention. It was a marketing decision IBM made when creating its first personal computer, the IBM PC.

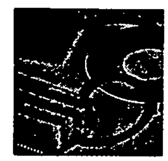
IBM wanted to keep the price down, and so it decided to build the computer from components that were already available, off the shelf, from several suppliers. IBM also made the overall design of the PC freely available to competitors. The only part of the machine IBM copyrighted was the BIOS, the crucial basic input/output system, computer code residing in a single chip that defined how software was to interact with the PC's hardware. The competition could create their own PCs as long as they duplicated the operations of IBM's BIOS without directly copying it.

While Apple continued to keep its design proprietary, IBM's openness encouraged the creation of IBM clones that could use the same software and hardware add-ons the PC used. And the clones, while competing with IBM, at the same time helped establish the IBM architecture as the machine that software and add-on hardware developers would design for. Precisely because the IBM PC was an evolutionary rather than revolutionary creation, it was able to create the critical mass needed to bring personal computers into every office and home.

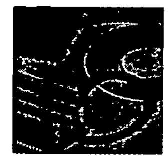
Index

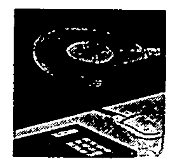
Symbols

A

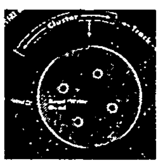


C



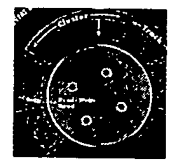


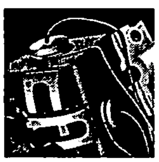
E

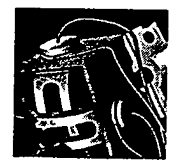


F

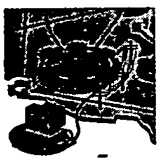
G







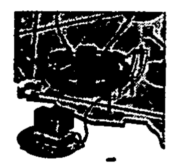
J



N

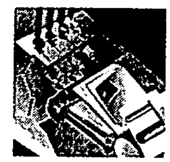
O

P

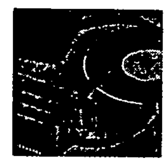


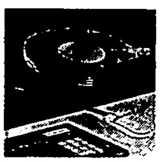
Q

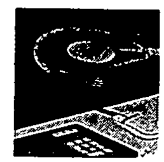
R



S





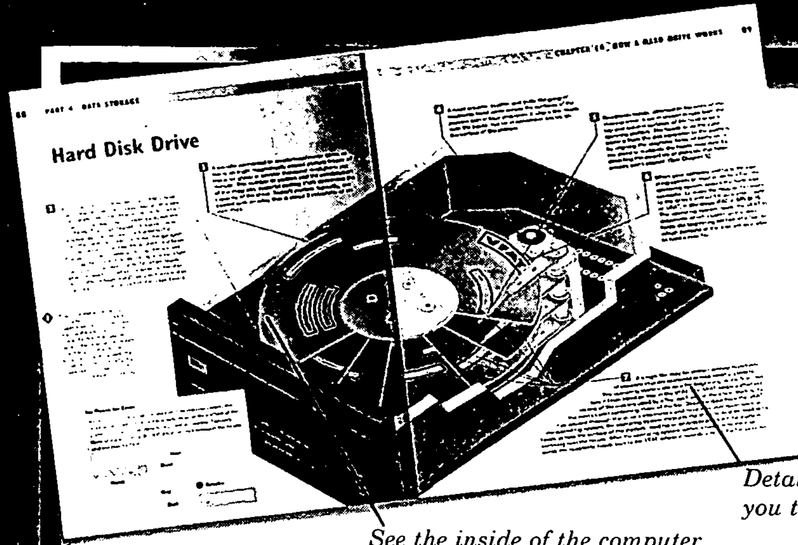


X

Y

Z

As Seen in *The New York Times*



See the inside of the computer and its components.

Detailed explanations walk you through the technology.

"If knowledge is the light of the world, then the contents of *How Computers Work* will certainly illuminate much about the inner workings of today's home and business computers without requiring the user to look under the hood."

—The New York Times

Ron White is a senior editor at *PC Computing*, where he writes the How It Works features and the Freeloader column. Founder of one of the earliest PC user groups, he has been writing about computers for 20 years. The second edition of *How Computers Work* won an award for best non-fiction computer book from the Computer Press Association.He can be reached via email at ron_white@zd.com. Also visit the companion Web site at www.Howcomputerswork.net.

SPECIAL MILLENNIUM EDITION INCLUDES:

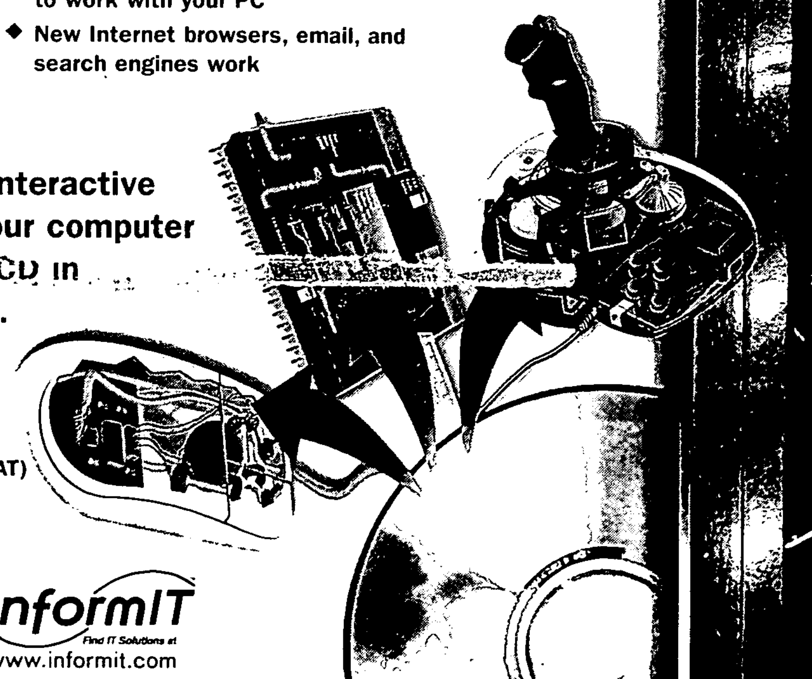
- **Pentium III and MMX processors**
- **New technologies such as fingerprint and voice recognition**
- **Notebook and palm computers**
- **MP3 music and digital audio**
- **Visions of how computers will work in the third millennium**
- **How virtual reality equipment adds a third dimension to PC worlds**
- **How force feedback controllers, accelerated graphics, and color printing create new ways to work with your PC**
- **New Internet browsers, email, and search engines work**

0 29236 72112 1

ISBN 0-7897-2112-0

9 780789 721129

Take an interactive tour of your computer with the CD in this book.



$29.99 USA / $44.95 CAN / £21.99 Net UK (inc of VAT)

Category	PCs/Hardware
Covers	General Computer Technology
User Level	Beginning—Intermediate

QUE®

www.quecorp.com

This Page is Inserted by IFW Indexing and Scanning Operations and is not part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☐ FADED TEXT OR DRAWING
- ☐ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☐ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☒ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: ______________________

IMAGES ARE BEST AVAILABLE COPY.
As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.